CONNEXIONS

The Interoperability Report

September 1987 Volume 1, No. 5

ConneXions - The Interoperability Report tracks current and emerging standards and technologies within the computer and communications industry.

In this issue:

ConneXions is published by Advanced Computing Environments, 21370 Vai Avenue, Cupertino, CA 95014, USA 408-996-2042.

ISSN 0894-5926

From the Editor

This issue is being released at the ISO Development Seminar in Monterey, California. Our main feature this month is a status report on the work being done by ISO to develop a set of international standards for Open Systems Interconnection. The article, entitled "OSI and the Seven Layers", is written by one of the seminar speakers: Sue Lebeck from the University of Wisconsin.

John Romkey is back again with another inside look at protocol implementations. This time he examines FTP and its "tiresome problems".

The Network Management Working Group is making progress in its effort to develop a set of RFCs. We asked one of the participants, Eric Benhamou from Bridge Communications to give us the "vendor viewpoint" on this effort.

Our guest editorial is by Einar ("Stef") Stefferud, long time player in the Internet community and participant in the IFIP and NBS standardization efforts. His "Plea for Internet Peace" was first delivered at the last TCP/IP Interoperability Conference and we asked him to write it up for a wider distribution.

Things are happening in the protocol testing arena. Unisys is under contract to the Defense Communications Agency to develop a prototype laboratory and software tools to support the installation of multiple protocol certification laboratories. We will bring you more details about this and other testing programs in subsequent issues of *ConneXions*.

Our next conference is the Second TCP/IP Interoperability Conference which will be held December 1-4, 1987 at the Hyatt Regency Crystal City in Arlington, VA. For complete details, including conference program and registration, contact us on 408-996-2042.

OSI and the Seven Layers

by Sue K. Lebeck, University of Wisconsin

The well-known DARPA-sponsored suite of protocols, TCP/IP and the corresponding application protocols, SMTP, FTP, and Telnet have been implemented by a wide variety of persons and groups for hardware and software systems representing a wide array of computer vendors. Operating over an extensive internet including the Arpanet, the NSFNET backbone and regional networks, various DoD networks, and a large number of local area networks for up to five years, these protocols have been the basis for a powerful, far-reaching, multi-vendor electronic communications system. The "DARPA Internet", as this networking system is called, has become an invaluable, even indispensable vehicle for research and government within the U.S. and selected European countries.

The need for OSI

Increasingly, the need has grown for a similar but globally-scaled system, international in origin and in outlook, which will exploit the lower layer backbones existing in countries other than the U.S. (notably Europe), as well as the infrastructure in place within the U.S. This system must support the traditional messaging, file transfer, and remote login applications, and be extendible to eventually support a rich and varied array of new distributed electronic applications.

The Open Systems Interconnection (OSI) effort of the International Standards Organization (ISO) addresses this need for a comprehensive international electronic interchange system. The interconnection of "open systems" -- systems which are internally independent, but which provide common external services via standardized publicly-defined protocols -- is the subject of the OSI seven-layer Reference Model.

OSI Reference Model

ISO developed the OSI seven-layer Reference Model as a formal approach for defining and viewing communication functions between open systems. This Basic Reference Model is defined in the ISO standard document ISO 7498, parts 1-4.

APPLICATION
PRESENTATION
SESSION
TRANSPORT
NETWORK
DATA LINK
PHYSICAL

Each layer in the OSI Reference Model is defined in terms of "services" and "protocols". A "service" defines the functionality which must be provided by a particular layer. Each layer utilizes the services of the layers beneath it, and offers an enhanced service to the layer above it. The implementation, or realization of a layer service within an open system is a local matter. A "protocol" defines the data unit formats to be exchanged between two open systems in order to effect the layer service. It also defines the proper usage and interpretation of these data units.

Throughout this article, the term "protocol" will be used to indicate both service and protocol, unless context suggests otherwise.

OSI Protocols

ISO has worked to develop a set of official standard protocols for use within each of the seven layers of the OSI model. These standard protocols are the result of a sometimes lengthy develop/propose/vote cycle. Proposed standards begin as "Draft Proposals", or DPs. As the development cycle repeats, the emerging intermediate standards progress through the status of DP-2, DP-n, then on to "Draft International Standard" (DIS), DIS-2, DIS-n, and finally to "International Standard" (IS). These ISO protocol standards become the road map of the OSI implementor. It is important to note that implementation of standards sometimes begins before their specifications have reached IS status.

Often, the initial versions of proposed standards have their beginnings within other standardization bodies. In particular, "Recommendations" developed by the International Consultative Committee for Telephone and Telegraph (CCITT) have frequently become the basis of ISO International Standards. This phenomenon is largely the result of a concerted effort between ISO and CCITT to develop compatible standards.

The following two sections list and very briefly describe the OSI protocols officially adopted by ISO for use within each of the seven layers, and cite the standards documents specifying these protocols. The sections are organized as "The Lower Layers" and "The Upper Layers".

The Lower Layers

The "lower layers" of the OSI model consist of the Physical, Data Link, Network, and Transport layers.

Physical Layer

The Physical layer is responsible for the encoding, decoding, and transmission of bits across a physical medium such as coaxial cable, twisted pair wire, fiber optic link, satellite, or microwave. ISO has defined Physical layer standards addressing such things as the mechanical characteristics, electrical characteristics, and signal quality of physical links.

Data Link Layer

The Data Link layer provides the ability to transmit electronic data reliably across a single physical link. The Data Link service is defined by DIS 8886.

continued on next page

OSI and the Seven Layers *(continued)*

Other ISO standards at the Data Link layer include a set of standards for High Level Data Link Control procedures (HDLC), among them ISO 3309 and ISO 4335. A particular class of HDLC, called Link Access Procedure Balanced (LAPB) by CCITT, is used within the data link sublayer of the widely-used X.25 standard. X.25 is the collective name of a set of CCITT standards for the Physical, Data Link, and Network layers.

Additionally, ISO's DIS 8802 series of standards defines Data Link protocols for use within local area networks. DIS 8802, part 3 for instance, defines the well-known Carrier Sense Multiple Access/ Collision Detection (CSMA/CD) protocol. The series also includes protocols for the Token Bus (DIS 8802/4), The Token Ring (DIS 8802/5), and the Slotted Ring (DIS 8802/7). This ISO 8802 series of standards corresponds to the IEEE 802 series of standards.

Network Layer

The Network layer provides Transport layer entities with an interface to the underlying subnetwork. It is concerned with routing packets to destination Transport entities on remote systems (via intermediate systems if necessary) and with controlling packet congestion within the subnetwork.

The Network service is defined by ISO 8348. The main body of this standard defines the connection-oriented network service; an addendum (AD1) defines the connectionless network service. The treatment of the connectionless network approach within an addendum is indicative of the connection-oriented bias of the original OSI designers. ISO 8348 also includes an addendum (AD2) on Network layer addressing; this addendum describes several different formats for OSI addressing.

X.25 Packet Layer Protocol

The X.25 packet layer protocol is the ISO standard for the connection-oriented approach. X.25 is heavily used over public data networks. European public data networks, run by national PTTs, provide X.25 service; this is the primary subnetwork backbone available in Europe.

Supporting standards for X.25 include DIS 8878, which describes the use of X.25 to provide a connection-oriented network service. Another standard, ISO 8208, describes the 1984 X.25 packet level protocol. This 1984 version of X.25 contains important enhancements over the 1980 version, most notably the address extension facility which enables OSI addresses to be specified during connection establishment. 1980 X.25 allows only for the use of X.121 addresses (defined by the CCITT X.121 Recommendation); this creates a problem, as today's OSI market provides primarily 1980 X.25 implementations. An additional facility called a Subnetwork Dependent Convergence Protocol (SNDCP) has been devised to accommodate OSI addressing within 1980 X.25. The DIS 8878 standard mentioned above includes an annex describing the use of 1980 X.25 and SNDCP. Many OSI software developers utilizing X.25 are opting to make do with 1980 X.25 implementations and engaging in bilateral agreements to substitute for SNDCP, with the expectation that 1984 X.25 implementations will be available soon.

A pair of standards, DIS 8881, parts 1 and 2, describe the use of X.25 over local area networks using connectionless or connection-oriented Link Level Control (LLC). Proponents of this configuration can be found, for instance, in the United Kingdom.

Connectionless Network Protocol

The Connectionless Network Protocol (CLNP, also known as ISO IP), is the ISO standard for the connectionless approach. The functional equivalent of DARPA's IP protocol, CLNP can be used to accommodate the concatenation of diverse subnetworks. In fact, CLNP can be operated over a subnetwork provided by multiplexed X.25 connections; it can also be operated over local area networks, the Arpanet, and any other sort of subnetwork. There are many proponents of the CLNP approach among those who have experienced the DARPA Internet.

Several ISO standards supporting the connectionless network service have been defined. These include ISO 8473, which defines the CLNP protocol; and DP 9542, which defines an end-system to intermediate-system (ES - IS) routing protocol for use with CLNP.

The above discussion correctly suggests that the ISO network standards allow for a large number of confusing options. Another document, DIS 8648, describes the various possible combinations.

Transport Layer

As with the Network layer, ISO has defined several flavors of service at the Transport layer. Happily, the options at this level are more straightforward. The appropriate choice depends on the capabilities provided by the underlying Network layer, and on the type of service (connection-oriented or connectionless) required by the next higher layer.

Five classes of Transport

Five flavors, or "classes" of connection-oriented Transport Protocol (TP) have been defined. Of these five classes (TP-0, TP-1, TP-2, TP-3, and TP-4), the classes TP-0 and TP-4 are the most commonly used. At one extreme, TP-0 provides little more than a transport-level access point to a connection-oriented network service. As such, it is typically used directly over X.25. At the other extreme, TP-4 provides an end-to-end reliable, sequenced, flow-controlled data path, making no assumptions about the reliability of the underlying network service. It is the functional counterpart to DARPA's TCP protocol, and is typically used over CLNP and a wide variety of subnetworks (easily including the subnetworks within the DARPA Internet).

The ISO standards defining connection-oriented TP are very stable. They include ISO 8072 describing the Transport service, and ISO 8073 describing the Transport protocol.

One connectionless Transport protocol is defined. This protocol is functionally the same as the DARPA UDP protocol, providing a transport access point and a fully checksummed datagram service.

The connectionless TP protocol is defined in DIS 8602; connectionless Transport service is defined in an addendum (AD1) of ISO 8072.

continued on next page

OSI and the Seven Layers *(continued)*

The Upper Layers

The "upper layers" of the OSI model consist of the Session, Presentation, and Application layers.

The OSI model divides the the application layer of the DARPA protocol model into three layers of functionality: session establishment and management functionality; presentation syntax management functionality; and application-specific functionality. This division results from the recognition (or, at least, the prediction) that many networking applications require similar high-level data transfer management facilities. Placing these facilities in a service layer below the application layer enables them to be shared by existing and future OSI applications.

Session Layer

The Session layer's task is to establish "sessions" between open systems, to provide a toolkit for managing data transfer over those sessions, and to police the use of those tools. Session tools include mechanisms to support half and full duplex data dialogues, data checkpointing, and session re-synchronization.

The Session service is defined by ISO 8326; the protocol is defined by ISO 8327.

A weighty consideration in the development of the Session layer standard was the European desire to support existing applications using Teletext. The protocol data unit formats used within Session were therefore designed to be compatible with the formats of T.62 commands and responses. Unable to begin from scratch, the Session protocol designers unfortunately inherited several cumbersome features from the T.62 standard.

Presentation Layer

The Presentation layer provides for an abstract view of electronic data types. It allows for two potentially dissimilar systems to negotiate the use of commonly understood abstract and concrete data syntaxes, and manages the subsequent use of those syntaxes during data transfer.

The Presentation layer service is defined by DIS 8822; the protocol is defined by DIS 8823.

An additional pair of Presentation layer standards provides the (currently) only example of an abstract syntax and a corresponding concrete transfer syntax. This first abstract syntax is appropriately entitled "Abstract Syntax Notation One", or ASN.1, and is defined in DIS 8824.2 (DIS-2). The concrete syntax, i.e. the encoding rules for the ASN.1 syntax, is defined by DIS 8825.2.

It is important to note that the service and protocol standards were completed relatively recently. Therefore, early OSI application specifications, such as 1984 MHS and early FTAM specifications (see below), omit the formal use of the Presentation service and instead use the Session service directly. The ASN.1 standard abstract and concrete syntax rules, however, are used even by these early OSI application specifications.

Application Layer

The Application layer of the OSI Reference Model contains the "real work" required by the end user. A wealth of standard applications for OSI are expected to emerge.

In OSI an application is viewed to consist of an "OSI portion", called an Application Entity (AE), and a "non-OSI portion". Using the most recent OSI terminology, an AE contains a "user-element" and one or more Application Service Elements (ASEs). The user-element defines the role of the particular open system. A particular ASE can be used within the context of various roles. Some ASEs are fairly application specific; others are used by a variety of applications.

The Association Control SE (ACSE) is one ASE intended for use by all standard ISO applications. ACSE is used to establish and release "associations" between AEs, and to establish a common understanding of the semantics ("application context") to be used over the associations. ACSE is defined by DIS 8649, part 2 (service) and DIS 8650, part 2 (protocol).

Currently, four applications, FTAM, MHS, VT, and JTM are defined; each will be described briefly below.

FTAM

"File Transfer, Access, and Management" (FTAM) is the first ISO application defined to utilize ACSE and the recently-specified Presentation service. The FTAM application provides for both file-level and record-level access to abstract structured files on remote open systems. It also provides for the update, transfer and management of remote files.

FTAM is specified in DIS 8571, parts 1-4. DIS 8571/1 provides a general description of FTAM; DIS 8571/2 defines the FTAM Virtual Filestore; DIS 8571/3 defines the FTAM service; and DIS 8571/4 defines the FTAM protocol.

Implementations of an agreed-upon FTAM subset have played an important role in multi-vendor testing, an important part of in the OSI development process. OSI vendors were able to develop implementations of this simplified FTAM quickly, making it available for interoperability testing of the lower layers.

Message Handling System

Another major ISO application is the "Message Handling System", or MHS. MHS provides a store and forward high-level messaging system. It is initially used to support the exchange of human-destined "interpersonal messages", but is generalized to support other types of messages. MHS provides a rich set of services, some "basic" and some "optional". It is multimedia in nature, designed to support a diverse set of "body types" in addition to ASCII, such as Teletex, telex, fascimile, and voice. The various body types are supported on a per-implementation basis.

X.400 series

MHS is defined in a set of ISO standards collectively referred to as "MOTIS": Message Oriented Text-Interchange System. These standards include DIS 8883 and DIS 9065, 9066, and 9072, parts 1-2. However, MHS is currently being implemented from the CCITT 1984 "Red Book" X.400-series of Recommendations, from which MOTIS is derived. These Recommendations are numbered X.400, X.401, X.408, X.409, X.410, X.411, X.420, and X.430. An enhanced version of MHS, available in 1988, will reflect a convergence effort on the part of ISO and CCITT.

continued on next page

OSI and the Seven Layers *(continued)*

The 1984 version of MHS was developed before the Presentation service and protocol were completed, and before the "ASE" concept for viewing OSI applications was developed. Therefore, its "Reliable Transfer Service" (RTS) portion, which is intended to handle actual message transfer functions, provides its own association control instead of using ACSE, and implements an *ad hoc* presentation protocol.

The 1988 version of MHS will use a version of RTS which performs only transfer functions (called RTSE), the common ACSE, and the Presentation service. A "pass-through" mode, using the *ad hoc* association control and presentation protocol, will be defined to accommodate the interworking of 1988 and 1984 MHS systems.

It is important to note that one of CCITT's MHS documents, Recommendation X.409, is mostly identical to and is the precursor of the ASN.1 presentation syntax standard.

MHS is a popularly implemented application; perhaps this is due to the global dependence on electronic mail that has evolved. Also, the MHS store-and-forward architecture lends itself for being "gateway"-ed to existing mail systems.

Other ISO applications

Currently less widely implemented ISO applications include "Virtual Terminal" (VT) and "Job Transfer and Management" (JTM). VT is defined by DIS 9040 (service) and DIS 9041 (protocol). JTM is defined by DIS 8831 (service) and DIS 8832 (protocol).

Directory Services

A very important ISO application whose specification is currently under development is the "Directory Services" application. This application will provide maintenance of and access to a database of name-to-attribute mappings. Attributes include addressing information, access control information, service information, and much more. Both ISO and CCITT have progressed work on Directory Services; they are currently working together to converge on one set of Directory Service documents. The ISO names for these convergence documents are DP 9594, parts 1-7.

It is intended that Directory Services will be used by all the OSI layers; the most imminent use of Directory Services will be by the 1988 version of MHS, to provide for such things as user-friendly naming and distribution list expansion.

Emergence of the Global ISO-net

The set of protocols delineated above comprises the ISO-"blessed" OSI protocol standards for which implementations are emerging from a variety of network software and hardware developers. OSI product seekers should look carefully when choosing OSI vendors. They must first of all distinguish between vendors offering products which merely follow the OSI Reference Model and those offering products which conform to the ISO standard protocols. Beyond that, they must be careful to obtain products that will interoperate effectively with other vendors' products. Interoperability problems derive from the newness and complexity of the ISO protocol suite.

NBS Implementors' Agreements

Many vendors are working hard to clarify ambiguities in the standards and to agree on the various implementation options, in the interest of interoperability. The National Bureau of Standards (NBS) has provided an arena for this agreement activity by coordinating regular "OSI Implementors' Agreements Workshops". Much has been accomplished within the scope of these workshops, providing an increased degree of confidence in the resulting implementations of these unproven protocols.

By choosing carefully, by supporting conscientious vendors, and by providing feedback on the emerging OSI products, OSI consumers can positively influence the development of OSI, expediting the reality of extensive electronic interaction with the ever-shrinking virtual world .

Hard work and good luck will make the ISO-net become an invaluable and indispensable vehicle at the global level, just as the DARPA Internet has been, for years, at the national level.

SUSAN K. LEBECK received an M.S. in Computer Science from the University of Wisconsin-Madison in 1983. Lebeck is an Associate Researcher in Network Communications at the University of Wisconsin-Madison. Her current work involves implementation of the ISO/CCITT X.400 Message Handling System for the IBM RT-PC workstation in a Berkeley UNIX environment. She has participated in the NBS OSI Implementor's Agreements Workshops and the IFIP WG 6.5 Working Conference on Message Handling. Previously, she spent two and a half years providing programming and user support for WISCNET, a DARPA protocol suite implementation for the IBM VM CMS system.

FTP's Tiresome Problems

by John Romkey

This time around we're going to talk about some common problems with FTP, the standard TCP/IP File Transfer Protocol. One such problem is that many of the FTP implementations in common use today were written prior to the latest version of the FTP specification (RFC 959) and don't properly implement some of the commands. Another problem is that some FTP's don't conform precisely to the protocol specification, but there are some simple implementation techniques that can be applied allowing conforming FTP's to operate with non-conforming FTP's. Finally, we look at some nice extensions to FTP in the user interface .

A bit of background

Before proceeding, we should delve into the FTP protocol briefly. Actions within the protocol are simple request-response transactions using TCP as a transport mechanism. When you start an FTP client, it makes a connection with the desired server and waits for the server to send a greeting banner.

continued on next page

FTP's Tiresome Problems *(continued)*

Whenever the FTP client wishes to send a command, it sends a line of ASCII text to the server with the command in it. The command is a word of up-to-four characters like USER, with whatever arguments are appropriate. For instance,

```
USER romkey
```

The server then processes the command and sends back a response. The response starts with a three digit reply code. The first digit indicates overall success or failure of the command, and can be used to drive a finite state machine that runs the FTP client, if desired. The second and third digits refine the meaning of the reply somewhat.

When a command has to transfer a file or directory listing, the server opens a second TCP connection (the *data connection*) back to the client, and the file transfer takes place on this connection. This connection can be managed in a variety of ways which can be controlled from the FTP client.

Reply codes can indicate total success, total failure, intermediate success (we've got the data connection open, now we're going to see about transferring the file), intermediate failure (we couldn't open the data connection, oh well), and success which requires further action (such as requiring a password to login).

Problems with reply codes

The FTP RFC specifies which reply codes to return in event of certain errors. Not all FTP server implementations conform to this specification, however. Some return a generic failure code, or some other code that violates the specification. Because many such servers are already in use, FTP clients should, if possible, rely only on the first digit of the return code. There may be some specific applications that require more information than this first digit provides, but in most cases it should be sufficient.

A general principle to follow if you're writing a TCP/IP implementation, or perhaps for network software in general: Be generous in what kind of input your programs accept, even if it does vary from the specifications, as long as it makes some sense. Be strict in the output that your programs create and follow the specifications as closely as possible. Programs that are built with this idea in mind will more likely be able to function with somewhat erroneous programs. Vendors who provide such programs will have to spend less time delving into mysterious customer problems and dealing with irate network users.

PWD or XPWD?

The 4.2 Berkeley Unix FTP was released before the most recent FTP RFC. It provided some commands that were experimental but became standard with that RFC. Their internal protocol names were XMKD, XRMD, XPWD and XCUP, which allowed FTP clients to create and destroy directories, find out the name of the current directory, and change to the parent of the current working directory. As they were standardized, the X's prefixed to the names were dropped, and they became MKD, RMD, PWD and CDUP.

Many vendors of 4.2-derived systems have not updated the FTP programs to follow the new standard.

Possible fixes Thus any new implementations of FTP will not be able to use these commands with these older systems. There are a couple of simple ways around this problem for FTP client implementations. One is to try the standard form of the command; if it succeeds, fine. If it fails as an unknown command (and here the FTP server had better return the correct reply code!), try the old X-form.

Alternatively, the FTP client can try to figure out that the server is a 4.2 server by asking the user to tell it or asking a name server. The latter approach is somewhat dangerous, because the system could quite possibly be a 4.2 system running an updated server with the command names changed. A simple fix for servers is to accept both forms of the command name.

Features Very few FTP's provide the entire protocol; usually a number of commands are left out. A bare-bones FTP will let you login and get and put files and change the transfer type and that's about it.

Most commonly, FTP implementations will allow directory listings, provide the append, rename and delete file commands, and also provide wildcard transfer commands (called *mget* and *mput* in Berkeley UNIX, for *m*ultiple get and put).

Some of the features that very few FTP's provide are the EBCDIC file type, the Passive command, any transfer *mode* other than Stream, and any transfer *structure* other than File. Also, very few non-Berkeley UNIX FTP's support the commands that create and destroy directories. EBCDIC is unnecessary unless you want the client FTP machine to be able to store files locally in EBCDIC format.

The passive command allows the client to open the data TCP connection to the server, instead of having the server open the data connection to the client. Vendors should note that the Air Force ULANA specification requires FTP implementations to support the passive (PASV) command.

With a little imagination, there are lots of interesting capabilities you can add to an FTP user interface. Clients can easily backup entire subdirectories to other systems (recursive send). When generating default filenames, the user interface can attempt to strip off directory delimiters. It could also optionally upper or lowercasify filenames (if you're transferring files to a UNIX host it's usually desirable to have their names come out in lower case).

Conclusion With a little elbow grease, you can also provide an FTP that will work well against other implementations, even if they are slightly out of spec. The big trick is just to be strict about what your FTP generates, but to relax restrictions on what it's willing to accept.

JOHN ROMKEY received his B.S. in Computer Science from MIT in 1985. While there, he worked with Prof. Jerome Saltzer and Dr. David Clark for three and a half years on the PC/IP project, a popular public domain implementation of TCP/IP for IBM PC's. After graduation, he spent a year as a staff member with Dr. Clark, and moved on to help found FTP Software, Inc., where he was Director of Software Development until July 1987. Although he retains ties with FTP, he currently spends his time consulting, developing new network services for IBM PC's, and trying to write science fiction.

Network Management Standardization: a Vendor's Perspective

by Eric Benhamou, Bridge Communications

A threat to competitiveness?

As momentum builds up within the TCP/IP community towards better standardization of network management procedures, vendors of communications equipment must each assess and face the impact of this trend. A legitimate initial reaction is to perceive the threat element present in any process that limits or reduces product differentiability. After all, up until recently the mere presence of any network management feature (even proprietary) in a communications product was positioned as a key selling advantage against competitors. Wouldn't this advantage disappear in the uniformity of standardized network management protocols and procedures? A closer examination of the current state of the market and the industry suggests that this reaction is ill-founded:

Many incompatibilities

1) The rapid maturation of the TCP/IP marketplace in the past two years, combined with that of the overall LAN industry has caused all leading vendors to increase their focus on network management products. These products or components tend to have similar base features. Inter-vendor incompatibilities are far greater than their differentiations and the possible competitive advantages that any vendor could derive from them. Collecting and reporting error and traffic statistics, for example, is quickly becoming a hygiene function rather than a unique extra.

Network management needed

2) The inevitable process of multivendor integration on customer premises is being hampered by the lack of network management standardization, which dampens the fundamental benefits of homogeneous TCP/IP protocols. Large network deployments are slowed down by the difficulties of network management integration. Vendors must sometimes resort to site-specific bilateral agreements or joint developments in order to achieve partial standardization. The market as a whole falls short of its full potential as large multivendor network procurements get deffered until broader standards come about.

What to standardize?

3) A vendor's needs for differentiation can be appeased. Standardization should not be misconstrued to mean uniformity. To start with, not every aspect of network management should nor could be standardized. Specifically, elements such as application programs, analysis tools to exploit the statistical network management data collected, user interfaces providing the access to network management function, and the presentation of network management information do not fall within the scope of a protocol standardization effort. Additionally, the broad spectrum of price/performance/quality that has served to differentiate TCP/IP implementations up to now will continue to apply when standardized network management protocols and procedures augment the basic stack. Finally, the diversity of equipment connectable to a TCP/IP network will require, if not guarantee, that proprietary extensions be provided and allowed for in any multivendor management standard.

Rationalization of the needs and benefits of network management standardization is a first step. A vendor's perspective is then affected by other factors:

One year window

1) The window of applicability for such a standard is bound in time. It starts now: the market for TCP/IP interoperable communications products is in the present. It ends at an indeterminate point in the mid-range future when the transition to OSI based networks is sufficiently underway to make an investment in TCP/IP network management less attractive. For all practical purposes, the window for the establishment of a standard ranges from a few months to about one year.

Migration to OSI

2) The OSI transition, however imminent it may be, must be reckoned with and planned for. Self-respecting vendors are cost-concious and reluctant to invest short-lived R&D money in an effort that does not bring them closer to an OSI product environment. Another requirement of a TCP/IP network management standard is to provide clear migration paths to OSI network management.

Pragmatic approach

3) The mechanisms by which standards come about within the TCP/IP community tend to be free-form and based upon individual contributions. The time pressure imposed upon the network management standard process forces an active involvement on the part of the vendors and a pragmatic, engineering-like approach on the part of individual participants of that process.

The TCP/IP marketplace has reached an interesting point of its maturation cycle where, caught between the momentum it has accumulated and the perspective of its obsolescence, it makes vendors develop a desire to agree, to extend its lifespan, and temporarily overcome their antagonisms and territorialities.

A Plea For Internet Peace

by Einar Stefferud, Network Management Associates

At the last Monterey Interoperability Conference, and at other conferences, I have detected a constant low level theme of conflict between advocates of TCP/IP and advocates of the new ISO standards. There appears to be some kind of mutual held notion that they must disagree, and that each must do what they can to inhibit the other's progress toward winning our common goal of Open Systems Interconnection.

This is very unfortunate, since the real enemy is to be found elsewhere, lurking among the proprietary standards that frustrate our efforts to get our tools to interoperate. I am writing this to plead for peace.

Contributions from the Internet side

If ISO is as inevitable as many believe, then why not strive to maximize its potentials? Why don't we find ways to contribute what the TCP/IP community has learned from its internetworking experience. Why take pleasure when Dave Clark (Chairman of the Internet Activities Board) observes that "The ISO community seems to be making the same mistakes that the TCP/IP community made, only five years later"? Yes, I caught myself chuckling too! But, as they say, it only hurts when we laugh.

continued on next page

A Plea For Internet Peace *(continued)*

On the other side, why should the ISO community feel that it would be helpful if TCP/IP would simply go away? Why should the ISO community be so aloof? How might we change things to gain the best of both worlds? What magic might we use? What stands in the way of cooperation?

Migration to ISO inevitable

Much has been said about how the ISO/CCITT standards are only different because of petty demands of international politics. I can generally agree with this assessment, but it is vastly over-simplified. In the long term we all desparately need an international industrial internet, and the current members of the TCP/IP community will need to belong to this global internet. There is no long term profit to be gained from isolation. Thus migration toward ISO is inevitable.

However there are real short term profits to be gained from continued use of the TCP/IP Internet protocols while we wait for the promised International Industrial Internet to materialize and to mature.

War and Peace

So, let's review what is at stake. What values can we gain with peaceful cooperation, and what values can we lose if we war instead? It only takes one side to start a war, but it always takes two or more sides to stop it. I have been working in both communities since 1979, and I have become more active over the past year to see what I can do about the conflicts that have arisen. It has been discouraging to watch people on both sides treat each other with disdain, but there have also been a few bright spots to fuel my hopes for peace.

Testing OSI ideas on the Internet

First, we should want to jointly take advantage of what has been learned about open systems and internetworking from our operational internet laborartory. We should also want to take advantage of this laboratory to conduct useful experiments and tests of new OSI ideas. The existing TCP/IP internet provides vital sanity checking facilities.

Advanced functionality in ISO

Second, we should want to capture the potential values that derive from the more advanced functionality and features of the ISO/CCITT upper layers, namely the tools provided by ASN.1 (Abstract Syntax Notation 1), ROSE (Remote Operations Service Entity), and ACSE (Association Control Service Entity). The ISO/CCITT application layer structure provides standard ways for distributed processes to interwork, regardless of local option choices for representation of information or invocation of processes. This offers a key advance in the whole internet concept. There is nothing in the current TCP/IP suite that can provide equivalent upper layer functionality, and there does not appear to be any effort, or reason to apply effort, to replicate these tools in the TCP/IP suite. Given the support behind ISO/CCITT, and the momentum that they are gaining, it appears best to go with the flow and work to get the new functionality.

So, what stands in the way? Why can't we just pick up our marbles and move on to the new game. I see several reasons. One is the matter of work styles in the two communities.

Differences in working style

The TCP/IP community is in the habit of letting individuals try out new ideas, and then let the better ideas progress toward acceptance. At least that is how we like to think it works, though it is not clear that every good idea progresses easily on its merits.

The ISO/CCITT community thrives on written paper based "contributions" with formal organizations and procedures for progressing them. These written contributions may be more theoretical and appear untested by comparison to TCP/IP community results. The ISO/CCITT community does not have access to an operational internet. It has no realistic choice other than to use their own long standing procedures, processes and organizations. I have been amazed on several occasions to see how much vitality a written contribution has in these ISO/CCITT processes. Written contributions simply take on a life of their own. Much more so than RFCs do in the TCP/IP Internet.

The Internet and ISO/CCITT modes of work progression do not readily mix. So, which is better? Which has produced the better results? Well, each has produced the better results in their own domain. TP4/IP and the whole Internet Catenet concept is unlikely to have progressed very far in the ISO/CCITT community without the ARPA Internet experiments. And, on the other hand, why do we see ASN.1, ROSE, & ACSE developing in ISO/CCITT? Why do SMTP, FTP, and Telnet each use their own "private ASN/ROS/ACS" protocols? I suspect environmental impacts are the cause.

I don't see much value in trying to choose which work style is best. We have both. We need both. And we need to find ways to make them work together while exploiting both styles. There is no hope of installing a global internet without first establishing agreement on the protocols. This is contrary to the basic ARPA Internet model which calls for stringing wire, attaching devices, and running experiments first.

International politics

National interests (ours and theirs, for any we and any they) inhibit basic protocol experiments across national boundaries. TCP/IP progress was well served by being sponsored by a single agency that could avoid broad democratic processes, while fostering collegial give and take within a small community. We cannot progress toward an International Industrial Internet without using democratic processes which always bring politics to the fore. The price of agreement is politics. We have to find ways to accomodate national interests by engaging in the arcane diplomatic negotiation processes of international standards activities. This is the only way to achieve our international internet goals.

CONNEXIONS
21370 Vai Avenue
Cupertino, CA 95014